

REMARKS

Claims 1 – 17 are pending in the application. Claims 1 – 17 stand rejected under 35 U.S.C. 101 and 35 U.S.C. 102 (b). Claims 1 – 17 include independent claims 1, 4, 7, 10, 13, and 16.

The Applicant respectfully disagrees with the Examiner's rejection of all claims under 101 as noted in the remarks below, particularly as the rejection relates to independent claim 13 and 16 and claims depending there from as they are method claims and not data structure claims as the Examiner indicates. Independent claims 1, 4, 7, 10, 13 and 16 have been amended for clarity.

With regard to the rejection under 102 (b), the Sanderson patent publication does not anticipate all of the recited claim limitations as noted herein. Specifically Sanderson does not anticipate, teach or suggest, integrated frameworks that revolve around data types where various applications can utilize the frameworks while maintaining commonality with regard to presentation, user interaction, and functionality. More specifically each of independent claims 1, 4, 7, 13, and 16 recite a UI element defining a data type and recite other frameworks that are configured or operate based on the data type. Independent claim 10 recites a tool for building the integrated frameworks.

Further, Sanderson is unrelated in that it is primarily a browser application tool that allows the browser application or other application to interface with the UI by way of work tasks that define widgets and that attempts to address the paradox between choosing a simple, low

bandwidth, low maintenance HTML based User Interface (UI) application or a more robust, high bandwidth, high maintenance UI application. Sanderson also attempts to resolve the problems with an intermediate solution of using applets (little applications) that can be shipped over from a server application and can be executed by a browser at run time to provide a more robust UI. Sanderson teaches that use of the applets simply increases the bandwidth problem.

Sanderson teaches a declarative UI generator to resolve the noted problems, which generates UI through the transmission and interpretation of context specifying configuration and workflow data. It is this transmission and interpretation that bridges the UI and the application logic. The declarative nature of the Sanderson UI generator simply indicates that the UI generator simply describes what tasks are to be performed and not how each task should be performed. The configuration data can be utilized to configure communicative links between the content servers and the data sources, and the workflow data can be utilized to describe tasks to be performed and based on these tasks suitable UI widgets can be selected and arranged. The described tasks can be parsed into units of work, which correspond to a pre-configured computing process and identifies the meta information data to be accessed. Refer to Sanderson at paragraphs [0012] through [0015]; [0026] through [0028]; [0037]; and [0048]. A widget is a control or is an interface element that a computer user interacts with, such as a window or a text box. Some typical examples of widgets are 'selection widgets' such as for example toggle buttons; 'navigation widgets'; and 'text input' edit fields.

However, the work tasks and related work descriptions for widget selection taught by Sanderson which allows the UI to interface with the application without applets and the like does

not anticipate, teach or suggest UI elements or frameworks that revolve around data types or logical data types whereby the frameworks can be used by various applications while maintaining commonality with regard to presentation, user interaction, and functionality. For example, a UI element can have a data type 'Currency'. The type 'Currency' can refer to an application that controls behavior and functionality. The behavior and functionality and appearance and user interaction will be the same whether the application is an Accounts Receivable application or a Billing Application. Sanderson does not teach integrated frameworks that revolve around data types where various applications can utilize the frameworks while maintaining commonality with regard to presentation, user interaction, and functionality. Specifically, independent Claims 1, 4, 7, 13, and 16 recite a UI Element defining a data type and varies other frameworks that operate based on the datatype.

The work tasks of Sanderson provides descriptions for a selection and positioning of widgets, which is a control or is an interface element that a computer user interacts with, such as a window or a text box. Sanderson does not teach a system of integrated frameworks that revolve around data type whereby the data type of a UI element determines how data is bound, how data is controled and handled in the GUI component of a screen within an overall navigation framework. Sanderson does not teach the access and use of these frameworks by an object oriented application. See paragraphs [0020] through [0024]; [0046] through [0049] and [0055] through [0057] of the Applicant's specification.

The present application teaches a tool for a client-server environment where integrated repositories, or frameworks, have been defined whereby all UI data elements and fields are

conformed to fit within the frameworks so that the screens have a predictable appearance and data handling is predictable. It is this predictability that allows a software developer to quickly develop or modify a UI application in a client-server environment without performing rudimentary coding of software. Whereas, Sanderson teaches a different approach where the UI generator browser application allows the data to define the tasks and widgets.

The Examiner rejected claims 1 – 18 under 35 U.S.C. 101 indicating that the claimed invention is directed to non-statutory subject matter, specifically indicating that the claims have limitations that don't have functional interrelations; lack physical article; and lack a tangible result .

With regard to functional interrelation, there are clear functional interrelations between the frameworks that are recited in the claims. For example, independent claim 1 recites a first UI repository limitation having a UI element and a second data binding framework limitation operable to bind data to the UI element recited in the first UI repository limitation and the second data binding framework limitation is interrelated to the GUI component of the third GUI framework limitation. The screen repository is clearly interrelated to the GUI framework and the navigation framework by the GUI component and screen attributes. Each of the independent claims has similar interrelations between the limitations.

With regard to the lack of physical article cited by the Examiner, the Applicant respectfully disagrees, however, each of the independent apparatus claims have been amended to clarify that a physical article is being claimed.

With regard to lack of tangible result cited by the Examiner, the Applicant, again respectfully disagrees, however, each of the independent method claims have been amended to clarify that a tangible results exists in the claim language.

Further, Data Structure Claims sometimes referred to as *Lowry* Claims In re Lowry, 32 F.3d 1579, (Fed. Cir 1994) are clearly patentable subject matter and weight should be given to the functional interrelations of the data structures, in this case in a client-server system that can be accessed by various object oriented software applications executing in the client-server system. The limitations are recited as residing on and accessible to a client-server system and are functionally interrelated as noted above, which clearly indicates that they are accessible by graphical user interface applications executing in the client-server system. However, the Applicant has further amended claims 1, 4, 7 and 10 to further clarify this point. Therefore, the Applicant respectfully asserts that the rejection under 101 has been overcome based on this response.

Applicant requests that the amendments be entered as outlined herein. Applicant asserts that claims 1 – 17 are now in condition for allowance and requests that said claims be allowed to proceed to issuance base on this response. Applicant asserts that the amendments responsive to the 35 U.S.C 101 rejections don't necessitate additional searching and that the amendments responsive to the 35 U.S.C. 102 are consistent with the originally filed independent claim 7 relating to the operation of frameworks based on data type, therefore additional searching is not required. Applicant requests an Advisory action if the Examiner doesn't believe that the case is in condition for allowance.

Application of: Brent Roberts, et al  
Serial No.: 10/722,630  
Amendment B

An Examiner Interview was conducted on July 10, 2007, however, no agreement was reached between the Examiner and the Attorney for the Applicant other than potential amendments to overcome the 101 rejections relating to interrelation between limitations, lack of tangible result, and lack of physical article and those amendments discussed are reflected herein. Attorney for Applicant disagreed with the merits of the 101 rejection, however, presents the amendments to further prosecution. Attorney for Applicant noted at least one point of novelty relating to use of 'data type' and this issue is presented herein.

If any issue regarding the allowability of any of the pending claims in the present application could be readily resolved, or if other action could be taken to further advance this application such as an Examiner's amendment, or if the Examiner should have any questions regarding the present amendment, it is respectfully requested that the Examiner please telephone Applicant's undersigned attorney in this regard.

Respectfully submitted,

Date:

July 18, 2007



Mark E. Stallion  
Reg. No. 46,132  
Blackwell Sanders Peper Martin LLP  
720 Olive Street, Suite 2400  
St. Louis, MO 63101  
314-345-6000  
ATTORNEYS FOR APPLICANT